

METHOD AND APPARATUS FOR PROCESSING DIRECTORY METADATA IN A DISTRIBUTED FILE SYSTEM, AND DEVICE

TECHNICAL FIELD

[0001] The present disclosure relates to the field of big data and distributed file systems in computer technologies, and specifically to a method and apparatus for processing directory metadata in a distributed file system, and a device.

BACKGROUND

[0002] With the increase of data volumes, the application of distributed file systems is more and more widespread. Currently, in a distributed file system, an index cell of one device node only includes a single index shard. This index shard is used to store index information for all directory metadata of this device node. When reading from or writing data to this device node, it is necessary to load the entire index shard, which can easily lead to the reduction of reading/writing efficiency.

[0003] Therefore, how to improve the reading/writing efficiency of the distributed file system has become an urgent problem to be solved.

SUMMARY

[0004] The present disclosure provides a method and apparatus for processing directory metadata in a distributed file system, and a device.

[0005] According to a first aspect of the present disclosure, a method for processing directory metadata in a distributed file system is provided. The distributed file system includes a plurality of device nodes, the device node corresponds to a metadata cell and an index cell. The metadata cell includes metadata shards corresponding to directory metadata in the device

node, and the index cell includes a plurality of index shards; each index shard is configured to store index information of a portion of the directory metadata. The method includes:

in response to a processing request, determining and loading a first index shard corresponding to directory metadata indicated by the processing request, and determining a metadata shard corresponding to the directory metadata indicated by the processing request according to the first index shard, where the processing request represents an adjustment to directory metadata in a device node;

adjusting information in the determined metadata shard, and adjusting index information corresponding to the directory metadata in a second index shard related to the directory metadata indicated by the processing request.

[0006] According to a second aspect of the present disclosure, an apparatus for processing directory metadata in a distributed file system is provided. The distributed file system includes a plurality of device nodes, the device node corresponds to a metadata cell and an index cell. The metadata cell includes metadata shards corresponding to directory metadata in the device node, and the index cell includes a plurality of index shards; each index shard is configured to store index information of a portion of the directory metadata. The apparatus includes:

a determining unit, configured to, in response to a processing request, determine and load a first index shard corresponding to directory metadata indicated by the processing request, and determine a metadata shard corresponding to the directory metadata indicated by the processing request according to the first index shard, where the processing request represents an adjustment to directory metadata in a device node;

an adjusting unit, configured to adjust information in the determined metadata shard, and adjust index information corresponding to the directory metadata in a second index shard related to the directory metadata indicated by the processing request.

[0007] According to a third aspect of the present disclosure, an electronic device is provided, including:

at least one processor; and a memory communicatively connected with the at least one processor; where the memory stores instructions executable by the at least one processor, and the instructions are executed by the at least one processor to enable the at least one processor

to execute the method according to the first aspect.

[0008] According to a fourth aspect of the present disclosure, a non-transitory computer-readable storage medium having computer instructions stored thereon is provided, where the computer instructions are used to cause a computer to execute the method according to the first aspect.

[0009] According to a fifth aspect of the present application, a computer program product is provided. The computer program product includes a computer program, where the computer program is stored in a readable storage medium. At least one processor of an electronic device can read the computer program from the readable storage medium, and the at least one processor executes the computer program to cause the electronic device to perform the method according to the first aspect.

[0010] It should be understood that content described in this section is neither intended to identify key or important features of embodiments of the present disclosure, nor to limit the scope of the present disclosure. Other features of the present disclosure will become easy to understand through the following description.

BRIEF DESCRIPTION OF DRAWINGS

[0011] Drawings are used to better understand the solutions, and do not constitute a limitation to the present disclosure. Among them:

[0012] FIG. 1 is a schematic diagram of a first embodiment of the present disclosure.

[0013] FIG. 2 is a schematic diagram of a second embodiment of the present disclosure.

[0014] FIG. 3 is a schematic diagram of a third embodiment of the present disclosure.

[0015] FIG. 4 is a schematic diagram of a fourth embodiment of the present disclosure.

[0016] FIG. 5 is a schematic diagram of a fifth embodiment of the present disclosure.

[0017] FIG. 6 is a schematic diagram of a sixth embodiment of the present disclosure.

[0018] FIG. 7 is a schematic diagram of a seventh embodiment of the present disclosure.

[0019] FIG. 8 is a schematic block diagram of an electronic device that may be used to implement embodiments of the present disclosure.

[0020] FIG. 9 is a schematic structural diagram of a distributed file system that can

implement embodiments of the present disclosure.

DESCRIPTION OF EMBODIMENTS

[0021] Exemplary embodiments of the present disclosure are illustrated below in conjunction with the drawings, where various details of the embodiments of the present disclosure are included to facilitate understanding, and they should be regarded as merely exemplary. Therefore, those of ordinary skill in the art should realize that various changes and modifications may be made to the embodiments described herein without departing from the scope of the present disclosure. Likewise, for clarity and conciseness, descriptions of well-known functions and structures are omitted in the following description.

[0022] With the increase of data volumes, the application of distributed file systems is more and more widespread. Currently, a device node of a distributed file system usually includes two parts: a metadata cell (Dentry cell) and an index cell (Dtree cell). The index cell includes a storage area for writing directories. However, due to a limitation of the number of directories stored in the storage area, the full memory can only support up to one billion level directories, and cannot reach one hundred billion level files. Moreover, due to a limitation of this single storage area for storing the directories, the system can only achieve a million-level Queries-per-second (Queries-per-second, QPS) for path resolution, which cannot meet the demand of hundreds of billions level.

[0023] In order to further improve storage capacity and QPS of the distributed file system, the present disclosure proposes a new architecture mode of index cells, which is used to solve the problems of poor scalability and low QPS of the distributed file system caused by a single-point bottleneck of the existing Dtree. Specifically, in the present disclosure, the index cell is optimized, where N index shards will be created in advance when the file system is created. When creating a directory (mkdir), deleting a directory (rmdir), etc., it will be routed to a fixed shard according to the full path Hash, and then corresponding index information will be created or deleted in a corresponding index shard. For a modification path (mv), it is necessary to search and process all index shards. The operation of routing to a specific index shard according to the full path Hash to perform path resolution can effectively improve QPS. Additionally, the

sharded path storage mechanism can effectively improve storage capacity of files.

[0024] For example, taking a query for directory C as an example, the directory metadata included in the query request is the directory C, and a target path of the directory metadata is /a/b/c. Firstly, according to the full path /a/b/c, an index shard can be determined by a mode of full path Hash. Then, through a remote procedure call (Remote Procedure Call, RPC), this index shard can be queried to acquire a metadata code of the directory C. Finally, another RPC is invoked to access the metadata shard corresponding to the metadata code, the query of attribute information of the directory C is realized.

[0025] The present disclosure provides a method and apparatus for processing directory metadata in a distributed file system, and a device, applied to the field of computer technologies, specifically big data and distributed file systems, to achieve the effect of improving the reading/writing efficiency of distributed file systems.

[0026] It should be noted that data in the embodiments is not targeted at any specific user and does not reflect personal information of any specific user. It should be noted that the data in the embodiments comes from public datasets.

[0027] In the technical solutions of the present disclosure, the collection, storage, use, processing, transmission, provision, and disclosure of user's personal information involved all comply with provisions of relevant laws and regulations and do not violate public order and morality.

[0028] In order to make readers understand the implementation principle of the present disclosure more deeply, the embodiment shown in FIG. 1 is further detailed below in conjunction with FIG. 2 to FIG. 5.

[0029] FIG. 1 is a schematic diagram of a first embodiment of the present disclosure. As shown in FIG. 1, the present disclosure provides a method for processing directory metadata in a distributed file system. The distributed file system used in the present disclosure includes a plurality of device nodes. The device node corresponds to a metadata cell and an index cell. The metadata cell includes metadata shards corresponding to directory metadata in the device node. The index cell includes a plurality of index shards. Each index shard is configured to store index information of a portion of the directory metadata. The method includes the following

steps.

[0030] 110, in response to a processing request, determining and loading a first index shard corresponding to directory metadata indicated by the processing request, and determining a metadata shard corresponding to the directory metadata indicated by the processing request according to the first index shard. The processing request represents an adjustment to directory metadata in a device node.

[0031] In this embodiment, a processing request of a user may be acquired. The processing request is used to adjust directory metadata in a device node. Optionally, the adjustment of the directory metadata may create the directory metadata, delete the directory metadata, and modify a path of the directory metadata. Optionally, the processing request may usually be an operation code. Optionally, the processing request needs to at least include path information and directory metadata to be processed. Optionally, when the processing request is used to indicate creating or deleting the directory metadata, the path information may at least include a target path. The target path is a path of the directory metadata to be deleted, or a path of the directory metadata to be created. Optionally, when the processing request is used to indicate modifying the path of the directory metadata, the path information may at least include an initial path and a final path. The initial path is a path of the directory metadata to be moved. The final path is a path that the directory metadata needs to reach after the moving.

[0032] According to the processing request, a first index shard corresponding to the directory metadata indicated by the processing request may be determined. Further, the first index shard is loaded, and in the first index shard, according to the path information contained in the processing request, index information corresponding to the path information is found through query in the first index shard. The index information may indicate a corresponding metadata shard, thereby determining the metadata shard corresponding to the directory metadata indicated by the processing request.

[0033] 120, adjusting information in the determined metadata shard, and adjusting index information corresponding to the directory metadata in a second index shard related to the directory metadata indicated by the processing request.

[0034] In this embodiment, a metadata shard may be determined according to step 110. The

metadata shard may be adjusted to realize creation and deletion of the metadata shard, as well as modification of information in the metadata shard. Optionally, the information in the metadata shard may be attribute information of the directory metadata corresponding to the metadata shard. In addition, according to the processing request, one or more second index shards related to the directory metadata of the processing request may also be determined in the index cell. After the second index shard(s) is determined, the index information in the second index shard(s) may be modified to ensure consistency of data in the index cell and the metadata cell. Optionally, in the process of modifying the metadata shard and the second index shard(s), the processing request is successfully executed only when modifications of all metadata shards and second index shards corresponding to the processing request are completed successfully.

[0035] In this embodiment, by determining and loading the first index shard corresponding to the directory metadata indicated by the processing request, the determination of the metadata shard can be realized, thereby realizing the modification of the metadata shard. This process achieves partial loading of data in the index cell by loading the first index shard, which reduces the data to be loaded and improves loading efficiency. Additionally, by modifying the index information in the second index shard(s) related to the directory metadata indicated by the processing request, the corresponding modification of the index shard is realized, which enhances modification efficiency of the data in the index cell and ensures consistency of the data in the index cell and the metadata unit.

[0036] FIG. 2 is a schematic diagram of a second embodiment of the present disclosure. As shown in FIG. 2, the present disclosure provides a method for processing directory metadata in a distributed file system. The method includes the following steps.

[0037] 210, calculating an index code according to path information of directory metadata. A value of the index code represents a code of an index shard.

[0038] In this embodiment, path information for the directory metadata may first be acquired from the processing request. An index code may be calculated based on the path information. The index code is the code of the index shard. Index codes correspond to index shards one by one.

[0039] Optionally, the information contained in the path information varies for different

processing requests, so the calculation process of the index code varies for different processing requests.

[0040] Optionally, when the processing request indicates deletion or creation of the directory metadata, the path information of the directory metadata includes a target path. At this time, it is necessary to calculate an index code of the directory metadata and an index code of a parent directory of the directory metadata. The calculation process may include following steps.

[0041] A), calculating a first index code according to the target path.

[0042] In this embodiment, the first index code of the directory metadata may be calculated based on full path information of the target path. The index shard indicated by the first index code corresponds to the path of the directory. That is, when the processing request indicates deletion of the directory metadata, the target path is included in the index shard indicated by the first index code. When the processing request indicates creating the directory metadata, the index shard indicated by the first index code is the index shard storing the target path.

[0043] B), determining a target parent path of a parent directory of the directory metadata in the target path according to the target path of the directory metadata.

[0044] In this embodiment, the parent directory of the directory metadata is determined according to the target path of the directory metadata. The target parent path of the parent directory is a path that remains after the directory corresponding to the directory metadata in the target path is deleted. For example, when the target path is /a/b/c, the target parent path is /a/b.

[0045] C), calculating a second index code according to the target parent path.

[0046] In this embodiment, the second index code of the directory metadata may be calculated based on full path information of the target parent path. The index shard indicated by the second index code corresponds to the target parent path. That is, the target parent path is included in the index shard indicated by the second index code.

[0047] Through the above steps A to C, the calculation of index codes is realized when the processing request indicates deletion or creation of directory metadata. At this time, the index codes may include a first index code for indicating an index shard where a path of the directory

metadata is located, and a second index code for indicating an index shard where a path of a parent directory of the directory metadata is located by the user. The calculation of the two index codes helps identify the index shards that need to be loaded, thus avoiding the need to load the entire index cell, reducing the amount of data loaded and improving data loading efficiency.

[0048] Optionally, when the processing request indicates modification of the path of the directory metadata, the path information of the directory metadata includes an initial path and a final path. The initial path indicates a path of the directory metadata before the modification, and the final path indicates a path of the directory metadata after the modification. At this time, it is necessary to determine, according to the initial path, an index code of the parent directory where the directory metadata is located before the modification. In addition, an index code of the parent directory where the directory metadata is located after the modification is determined according to the final path. The calculation process may include:

D), determining a first parent path of a parent directory of the directory metadata in the final path according to the final path of the directory metadata;

E), calculating a third index code according to the first parent path;

F), determining a second parent path of a parent directory of the directory metadata in the initial path according to the initial path of the directory metadata;

G), calculating a fourth index code according to the second parent path.

[0049] Implementations of the above steps D and F are similar to that of the above step B, and implementations of the above steps E and G are similar to that of the above step C, which will not be repeated here. Through the execution of the above steps D to G, when the processing request indicates modification of the path of the directory metadata, the acquisition of the path of the parent directory of the directory metadata on the final path and the initial path, as well as the calculation of the index codes corresponding to the two parent directories are realized. The calculation of the two index codes helps identify the index shards that need to be loaded, thus avoiding the need to load the entire index cell, reducing the amount of data loaded and improving data loading efficiency.

[0050] Optionally, taking the target path as an example, the specific process of calculating

the first index code according to the target path may include the following steps.

[0051] H), calculating an MD5 value of the target path.

[0052] In this embodiment, the MD5 value may be calculated according to the target path. The MD5 value may be calculated using a preset message-digest algorithm.

5 **[0053]** I), determining the first index code according to the MD5 value and a preset number of the index shards in the index cell.

[0054] In this embodiment, after determining the MD5 value, a final index code is obtained by applying a modulo operation with a preset number of index shards in the index cell. Optionally, the number of index shards in the index cell may be set in advance when the device
10 node of the distributed file system is created. The modulo operation may realize allocation of the target path to the index shard. This allocation method can improve data balance and prevent index information from being concentratedly written to a limited subset of the index shards.

[0055] Optionally, the first index code and the second index code may be the same. And the third index code and the fourth index code may be the same.

15 **[0056]** 220, loading the first index shard corresponding to the directory metadata indicated by the processing request according to the index code.

[0057] In this embodiment, after determining at least one index code according to step 210, the first index shard may be loaded according to this index code.

[0058] In this embodiment, by determining and loading at least one first index shard
20 corresponding to the directory metadata indicated by the processing request, the loading of partial data in the index cell is realized, thus avoiding the need to load the entire index cell, reducing the amount of data loaded and improving loading efficiency of the index cell in response to the processing request. Furthermore, it is helpful to improve efficiency of path resolution in the distributed file system and increase an upper limit of QPS.

25 **[0059]** FIG. 3 is a schematic diagram of a third embodiment of the present disclosure. As shown in FIG. 3, the present disclosure provides a method for processing directory metadata in a distributed file system. The method includes the following steps.

[0060] 310, determining a metadata code corresponding to directory metadata in an index shard according to path information of the directory metadata indicated by a processing request.

The metadata code represents a code of the metadata shard.

[0061] In this embodiment, the processing request includes path information of the directory metadata. According to the path information, a metadata code corresponding to the path information may be determined in the index shard. The metadata code is the code of the metadata shard. Metadata codes correspond to metadata shards one by one.

[0062] Optionally, the specific process of determining the metadata code corresponding to the path information in the index shard may include the following steps.

[0063] J), determining index information corresponding to the path information in the index shard according to the path information of the directory metadata indicated by the processing request.

[0064] In this embodiment, the path information corresponding to the directory metadata may be determined according to the second embodiment. According to the path information, the index information can be found through query in the directory metadata.

[0065] K), determining the metadata code corresponding to the index information in the index shard.

[0066] In this embodiment, a correspondence between the index information and the metadata code may be stored in the index shard. According to this correspondence, the metadata code corresponding to the index information may be determined after the index information is determined.

[0067] Through the above steps J and K, a query can be conducted in the index shard according to the path information to obtain the index information and the metadata code corresponding to the index information, thereby realizing the determination of the metadata shard, which improves the efficiency of determining the metadata shard, and improve the efficiency of path resolution.

[0068] Optionally, when the processing request indicates deletion or creation of the directory metadata, the index codes calculated in the second embodiment may include a first index code and a second index code. The first index code is an index code corresponding to the target path of the directory metadata indicated by the processing request. A first index shard may be determined according to the first index code. In the first index shard, a search may be

conducted according to the target path to obtain index information of the target path, as well as a metadata code corresponding to the index information. The metadata code may point to a metadata shard of the directory metadata. The second index code is an index code corresponding to a target parent path of a parent directory of the directory metadata indicated by the processing request. Another first index shard may be determined according to the second index code. In this first index shard, a search may be conducted according to the target parent path to obtain index information of the target parent path, as well as a metadata code corresponding to the index information. The metadata code may point to a metadata shard of the parent directory of the directory metadata.

[0069] Optionally, when the processing request indicates modification of the path of the directory metadata, the index codes calculated in the second embodiment may include a third index code and a fourth index code. The third index code is an index code corresponding to a first parent path of a parent directory of the directory metadata in the final path. A first index shard may be determined according to the third index code. In the first index shard, a search may be conducted according to the first parent path to obtain index information of the first parent path, as well as a metadata code corresponding to the index information. The metadata code may point to a metadata shard of the parent directory of the directory metadata in the final path. The fourth index code is an index code corresponding to a second parent path of a parent directory of the directory metadata in the initial path. Another first index shard may be determined according to the fourth index code. In the first index shard, a search may be conducted according to the second parent path to obtain index information of the second parent path, as well as a metadata code corresponding to the index information. The metadata code may point to a metadata shard of the second parent path of the parent directory of the directory metadata in the initial path.

[0070] 320, determining the metadata shard corresponding to the directory metadata indicated by the processing request according to the metadata code.

[0071] In this embodiment, after at least one metadata code is determined according to step 310, the corresponding metadata shard can be acquired in the metadata cell according to the metadata code, thereby realizing the reading/writing of the metadata shard.

[0072] In this embodiment, a metadata code is determined by querying index information in a first index shard according to the path information, and the reading/writing of a metadata shard is realized according to the metadata code, thereby improving the efficiency of determining the metadata shard, as well as the efficiency of path resolution in the distributed file system.

[0073] FIG. 4 is a schematic diagram of a fourth embodiment of the present disclosure. As shown in FIG. 4, the present disclosure provides a method for processing directory metadata in a distributed file system. In this method, after a metadata shard is determined, information in the metadata shard may be adjusted. According to three different processing requests of deletion, creation and modification, the adjustment process of the metadata shard may include the following steps.

[0074] 410, when the processing request indicates deletion of directory metadata, deleting a metadata shard corresponding to the directory metadata and modifying a metadata shard corresponding to a parent directory of the directory metadata.

[0075] In this embodiment, when the processing request indicates the deletion of the directory metadata, two metadata shards can be acquired according to the second embodiment and the third embodiment. Among them, one metadata shard is a metadata shard corresponding to the directory metadata, and another metadata shard may be a metadata shard corresponding to the parent directory of the directory metadata. Because the directory metadata needs to be deleted in response to the processing request, the allocation of metadata corresponding to the directory metadata will become irrelevant significance after the deletion of the directory metadata, so the metadata shard corresponding to the directory metadata may be deleted accordingly. After the directory metadata is deleted, subdirectories associated with the parent directory of the directory metadata change. Therefore, in order to ensure that attribute information of the parent directory of the directory metadata matches an actual situation, it is necessary to update the information in the metadata shard of the parent directory of the directory metadata.

[0076] Optionally, the above-mentioned procedure for processing metadata shards may include the following steps.

[0077] L), deleting the first metadata shard.

[0078] In this embodiment, the metadata shard corresponding to the directory metadata is referred to as a first metadata shard. The first metadata shard may be deleted from this metadata cell.

5 **[0079]** M), subtracting a quantity of the directory metadata from information of the second metadata shard.

[0080] In this embodiment, the metadata shard corresponding to the parent directory of the directory metadata is referred to as a second metadata shard. The quantity of the directory metadata may be subtracted from the information of the second metadata shard. For example,
10 the information of the second metadata shard may include the number of subdirectories under the parent directory. After the deletion of the directory metadata, the number of subdirectories will be subtracted by one.

[0081] Through the step 410, when the processing request is used for deletion of the directory metadata, the metadata shards will be processed, thereby ensuring execution of the
15 processing request, improving the response efficiency of the processing request and improving user experience.

[0082] 420, when the processing request indicates creation of the directory metadata, creating the metadata shard corresponding to the directory metadata and modifying the metadata shard corresponding to the parent directory of the directory metadata.

20 **[0083]** In this embodiment, when the processing request indicates the creation of the directory metadata, two metadata shards can be acquired according to the second embodiment and the third embodiment. Among them, one metadata shard is a metadata shard corresponding to the directory metadata, and another metadata shard may be a metadata shard corresponding to the parent directory of the directory metadata. Because the directory metadata needs to be
25 created in response to the processing request, the metadata shard corresponding to the directory metadata needs to be created in this metadata cell. After the directory metadata is created, subdirectories associated with the parent directory of the directory metadata change. Therefore, in order to ensure that attribute information of the parent directory of the directory metadata matches an actual situation, it is necessary to update the information in the metadata shard of

the parent directory of the directory metadata.

[0084] Optionally, the above-mentioned procedure for processing metadata shards may include the following steps.

[0085] N), creating the first metadata shard.

5 **[0086]** In this embodiment, the metadata shard corresponding to the directory metadata is referred to as a first metadata shard. The first metadata shard may be created in this metadata cell.

[0087] O), increasing a quantity of the directory metadata to information of the second metadata shard.

10 **[0088]** In this embodiment, the metadata shard corresponding to the parent directory of the directory metadata is referred to as a second metadata shard. The quantity of the directory metadata may be increased to the information of this second metadata shard. For example, the information of the second metadata shard may include the number of subdirectories under the parent directory. After the creation of the directory metadata, the number of subdirectories will
15 be increased by one.

[0089] Through the step 420, when the processing request is used for creation of the directory metadata, the metadata shards will be processed, thereby ensuring execution of the processing request, improving the response efficiency of the processing request and improving user experience.

20 **[0090]** 430, when the processing request indicates modification of the directory of the directory metadata, modifying the metadata shard corresponding to the parent directory of the directory metadata indicated by the final path and the metadata shard corresponding to the parent directory of the directory metadata indicated by the initial path.

[0091] In this embodiment, when the processing request indicates the modification of the
25 directory of the directory metadata, two metadata shards can be acquired according to the second embodiment and the third embodiment. Among them, one metadata shard is a metadata shard corresponding to the parent directory of the directory metadata indicated by the final path, and another metadata shard may be a metadata shard corresponding to the parent directory of the directory metadata indicated by the initial path. The directory of the directory metadata will

be changed in response to the processing request, which means the parent directory of the directory metadata will be changed. That is, it is deleted from a parent directory corresponding to the initial path and added to a parent directory corresponding to the final path. From the perspective of the directory metadata per se, its subdirectories remain unchanged. Therefore,
5 the metadata shards of the directory metadata itself remain unchanged.

[0092] Optionally, the above procedure for processing metadata shards may include the following steps.

[0093] P), increasing a quantity of the directory metadata to information of the third metadata shard.

10 **[0094]** In this embodiment, the metadata shard corresponding to the parent directory of the directory metadata indicated by the final path is referred to as a third metadata shard. The quantity of the directory metadata may be increased to the information of this third metadata shard. For example, the information of the third metadata shard may include the number of subdirectories under the parent directory. After the directory metadata is moved to the parent
15 directory indicated by the final path, the number of subdirectories will be increased by one.

[0095] Q), subtracting a quantity of the directory metadata from information of the fourth metadata shard.

[0096] In this embodiment, the metadata shard corresponding to the parent directory of the directory metadata indicated by the initial path is referred to as a fourth metadata shard. The
20 quantity of the directory metadata may be subtracted from the information of this fourth metadata shard. For example, the information of the fourth metadata shard may include the number of subdirectories under the parent directory. After the directory metadata is moved away from the parent directory indicated by the initial path, the number of subdirectories will be subtracted by one.

25 **[0097]** Through the step 430, when the processing request is used for modification of a directory of directory metadata, the metadata shards will be processed, thereby ensuring execution of the processing request, improving the response efficiency of the processing request and improving user experience.

[0098] In this embodiment, by modifying the metadata shards, a response to the processing

request within the metadata cell is realized, thereby improving the response efficiency of the processing request and improving user experience.

[0099] FIG. 5 is a schematic diagram of a fifth embodiment of the present disclosure. As shown in FIG. 5, the present disclosure provides a method for processing directory metadata in a distributed file system. In this method, by determining a second index shard related to the directory metadata indicated by the processing request, the processing of the second shard may be realized in index shards. According to three different processing requests of deletion, creation and modification, the determining process and the adjustment process of the second index shard may include following steps.

[0100] 510, when the processing request indicates deletion of directory metadata, deleting index information corresponding to a target path in an index shard corresponding to the target path of the directory metadata.

[0101] In this embodiment, when the processing request indicates deletion of the directory metadata, the path information of the processing request may include a target path. The target path is a path of the directory metadata that needs to be deleted. According to the target path, the index code corresponding to the target path may be determined according to the method for determining an index shard shown in the second embodiment, so as to determine the second index shard. When the directory metadata is deleted, the index information of the directory metadata will become irrelevant. Therefore, the index information corresponding to the target path may be deleted from the index shard.

[0102] Optionally, the above procedures for determining and processing metadata shards may specifically include the following steps.

[0103] R), determining the second index shard according to the target path of the directory metadata.

[0104] In this embodiment, an MD5 value may be determined according to the target path of the directory metadata. Further, an index code is determined according to a modulus of a preset number of the index shards and the MD5 value. A corresponding second index shard is determined according to the index code.

[0105] S), deleting the index information corresponding to the target path of the directory

metadata from the second index shard.

[0106] In this embodiment, the index information corresponding to the target path may be obtained by querying the target path in the second index shard. The index information may be deleted from the second index shard. Optionally, the deletion of the index information includes deleting the metadata code which is stored in the second index shard corresponding to the index information. Optionally, the deletion of the index information includes deleting all information in the second index shard which is stored corresponding to the index information.

[0107] Through the step 510, when the processing request is used for deletion of directory metadata, the metadata shards in the index cell will be processed, thereby ensuring execution of the processing request, improving the response efficiency of the processing request and improving user experience.

[0108] 520, when the processing request indicates creation of the directory metadata, creating the index information corresponding to the target path in the index shard corresponding to the target path of the directory metadata.

[0109] In this embodiment, when the processing request indicates the creation of the directory metadata, the path information of the processing request may include a target path. The target path is a path of the directory metadata that needs to be created. According to the target path, the index code corresponding to the target path may be determined according to the method for determining an index shard shown in the second embodiment, so as to determine the second index shard. When the directory metadata is created, it is necessary to synchronously create the index information of the target path of the directory metadata in the index shard, so as to ensure that the newly created directory metadata has corresponding index information.

[0110] Optionally, the above procedures for determining and processing metadata shards may specifically include the following steps.

[0111] T), determining the second index shard according to the target path of the directory metadata.

[0112] In this embodiment, an MD5 value may be determined according to the target path of the directory metadata. Further, an index code is determined according to a modulus of a preset number of the index shards and the MD5 value. A corresponding second index shard is

determined according to the index code.

[0113] U), creating the index information corresponding to the target path of the directory metadata from the second index shard.

[0114] In this embodiment, the index information of the target path may be created in the second index shard. Optionally, after the creation of the index information is completed, a metadata code of the metadata shard of the directory metadata that has been created may also be acquired from the metadata shard. The metadata code and a correspondence between the metadata code and the index information are stored in the second index shard.

[0115] Through the step 520, when the processing request is used for creation of directory metadata, the metadata shards in the index cell will be processed, thereby ensuring execution of the processing request, improving the response efficiency of the processing request and improving user experience.

[0116] 530, when the processing request indicates modification of the directory of the directory metadata, deleting index information containing the directory metadata from each index shard, and creating index information of a path in an index shard corresponding to the path according to the modified paths of the subdirectory of the directory metadata and the directory metadata.

[0117] In this embodiment, when the processing request indicates the modification of the directory of the directory metadata, the path information of the processing request may include a final path and an initial path. The final path is a path after the modification of the directory metadata, and the initial path is a path before the modification of the directory metadata.

[0118] When the directory metadata is moved out of the initial path, all subdirectories of the directory metadata will be moved out together. The existing index information containing the directory metadata in the index cell will become invalid. Therefore, the index information containing the directory metadata may be deleted from each index shard.

[0119] After the directory metadata is moved into the final path, paths of subdirectories in the directory metadata will be modified accordingly. The index information of both the directory metadata and the subdirectory of the directory metadata need to be recreated. Therefore, according to modified paths of the directory metadata and its subdirectory, the index

information of the path may be created in the index shard corresponding to this path.

[0120] Optionally, the above procedures for determining and processing metadata shards may specifically include the following steps.

[0121] V), if the index shard contains the directory metadata, determining the index shard as the second index shard and deleting the index information containing the directory metadata in the second index shard.

[0122] In this embodiment, by traversing index information in each index shard, the determination of whether the directory metadata is contained in this index information can be realized. When the index shard contains the directory metadata, the index shard may be determined as the second index shard. When the index shard contains the directory metadata, the index information may be deleted.

[0123] W), determining modified paths of the directory metadata and its subdirectory according to the final path and the initial path.

[0124] In this embodiment, a subdirectory path of a subdirectory of the directory metadata before modification may be acquired. By replacing the portion of the directory metadata and its previous portion in the subdirectory path with the final path, the modified path of the subdirectory is obtained. The modified path of the directory metadata is the final path.

[0125] X), creating index information of a modified path in a corresponding second index shard according to the modified paths of the directory metadata and the subdirectory of the directory metadata.

[0126] In this embodiment, according to the modified paths obtained in the step W, the corresponding index shards may be determined in the index cell one by one, and the index information of the modified paths may be created in the index shards. Optionally, after the creation of the index information of the modified paths is completed, metadata codes of the directory metadata and the subdirectory of the directory metadata may be written into corresponding positions of the index information of the modified paths.

[0127] Through the step 530, when the processing request is used for deletion of directory metadata, the metadata shards in the index cell will be processed, thereby ensuring execution of the processing request, improving the response efficiency of the processing request and

improving user experience.

[0128] In this embodiment, a response to the processing request within the index cell is realized by modifying the metadata shards, thereby improving the response efficiency of the processing request and improving user experience.

5 **[0129]** FIG. 6 is a schematic diagram of a sixth embodiment of the present disclosure. As shown in FIG. 6, an apparatus 600 for processing directory metadata in a distributed file system is provided. The distributed file system includes a plurality of device nodes, the device node corresponds to a metadata cell and an index cell. The metadata cell includes metadata shards corresponding to directory metadata in the device node, and the index cell includes a plurality
10 of index shards; each index shard is configured to store index information of a portion of the directory metadata. The apparatus includes:

a determining unit 610, configured to in response to a processing request, determine and load a first index shard corresponding to directory metadata indicated by the processing request, and determine a metadata shard corresponding to the directory metadata indicated by
15 the processing request according to the first index shard. The processing request represents an adjustment to directory metadata in a device node;

an adjusting unit 620, configured to adjust information in the determined metadata shard, and adjust index information corresponding to the directory metadata in a second index shard related to the directory metadata indicated by the processing request.

20 **[0130]** The apparatus of this embodiment can perform the technical solutions of the above method embodiments, and the specific implementation processes and principles thereof are the same and will not be repeated here.

[0131] FIG. 7 is a schematic diagram of a seventh embodiment of the present disclosure. As shown in FIG. 7, an apparatus 700 for processing directory metadata in a distributed file
25 system is provided. The distributed file system includes a plurality of device nodes, the device node corresponds to a metadata cell and an index cell. The metadata cell includes metadata shards corresponding to directory metadata in the device node, and the index cell includes a plurality of index shards; each index shard is configured to store index information of a portion of the directory metadata. The apparatus includes:

a determining unit 710, configured to in response to a processing request, determine and load a first index shard corresponding to directory metadata indicated by the processing request, and determine a metadata shard corresponding to the directory metadata indicated by the processing request according to the first index shard. The processing request represents an
5 adjustment to directory metadata in a device node;

an adjusting unit 720, configured to adjust information in the determined metadata shard, and adjust index information corresponding to the directory metadata in a second index shard related to the directory metadata indicated by the processing request.

[0132] Optionally, the processing request includes path information of the directory
10 metadata. The determining unit 710 includes:

a first calculating module 711, configured to calculate an index code according to the path information of the directory metadata, where a value of the index code represents a code of the index shard;

a first determining module 712, configured to determine and load the first index
15 shard corresponding to the directory metadata indicated by the processing request according to the index code.

[0133] Optionally, when the processing request indicates deletion or creation of the directory metadata, the path information of the directory metadata includes a target path. The first calculating module 711 includes:

20 a first calculating submodule 7111, configured to calculate a first index code according to the target path;

a first determining submodule 7112, configured to determine a target parent path of a parent directory of the directory metadata in the target path according to the target path of the directory metadata;

25 a second calculating submodule 7113, configured to calculate a second index code according to the target parent path.

[0134] Optionally, the first calculating submodule 7111 is specifically configured to:

calculate an MD5 value of the target path; and

determine the first index code according to the MD5 value and a preset number of

the index shards in the index cell.

[0135] Optionally, when the processing request indicates modification of a path of the directory metadata, the path information of the directory metadata includes an initial path and a final path. The first calculating module 711 includes:

5 a second determining submodule 7114, configured to determine a first parent path of a parent directory of the directory metadata in the final path according to the final path of the directory metadata;

 a third calculating submodule 7115, configured to calculate a third index code according to the first parent path;

10 a third determining submodule 7116, configured to determine a second parent path of a parent directory of the directory metadata in the initial path according to the initial path of the directory metadata;

 a fourth calculating submodule 7117, configured to calculate a fourth index code according to the second parent path.

15 **[0136]** Optionally, the processing request includes path information of the directory metadata. The determining unit 710 includes:

 a second determining module 713, configured to determine a metadata code corresponding to the directory metadata in the first index shard according to the path information of the directory metadata indicated by the processing request. The metadata code
20 represents a code of the metadata shard;

 a third determining module 714, configured to determine the metadata shard corresponding to the directory metadata indicated by the processing request according to the metadata code.

[0137] Optionally, the second determining module 713 includes:

25 a first determining submodule 7131, configured to determine index information corresponding to the path information in the index shard according to the path information of the directory metadata indicated by the processing request;

 a second determining submodule 7132, configured to determine the metadata code corresponding to the index information in the index shard.

[0138] Optionally, when the processing request indicates deletion of the directory metadata, the metadata shard includes a first metadata shard corresponding to the directory metadata and a second metadata shard corresponding to a parent directory of the directory metadata. The adjusting unit 720 includes:

- 5 a first deleting module 7211, configured to delete the first metadata shard; and
- a first adjusting module 7212, configured to subtract a quantity of the directory metadata from information of the second metadata shard.

[0139] Optionally, when the processing request indicates creation of the directory metadata, the metadata shard includes a first metadata shard corresponding to the directory metadata and
10 a second metadata shard corresponding to a parent directory of the directory metadata. The adjusting unit 720 includes:

- a first creating unit 7213, configured to create the first metadata shard;
- a second adjusting module 7214, configured to increase a quantity of the directory metadata to information of the second metadata shard.

[0140] Optionally, when the processing request indicates modification of a directory of the directory metadata, the metadata shard includes a third metadata shard corresponding to a parent directory indicated by a final path of the directory metadata and a fourth metadata shard corresponding to a parent directory indicated by an initial path of the directory metadata. The adjusting unit 720 includes:

- 20 a third adjusting module 7215, configured to increase a quantity of the directory metadata to information of the third metadata shard; and
- a fourth adjusting module 7216, configured to subtract a quantity of the directory metadata from information of the fourth metadata shard.

[0141] Optionally, when the processing request indicates deletion of the directory metadata, path information of the directory metadata includes a target path. The adjusting unit 720
25 includes:

- a first determining module 7221, configured to determine the second index shard according to the target path of the directory metadata; and
- a first deleting module 7222, configured to delete the index information

corresponding to the target path of the directory metadata from the second index shard.

[0142] Optionally, when the processing request indicates creation of the directory metadata, path information of the directory metadata includes a target path. The adjusting unit 720 includes:

5 a second determining module 7223, configured to determine the second index shard according to the target path of the directory metadata; and

 a first creating module 7224, configured to create the index information corresponding to the target path of the directory metadata in the second index shard.

[0143] Optionally, when the processing request indicates modification of a directory of the
10 directory metadata, path information of the directory metadata includes a final path and an initial path. The adjusting unit 720 includes:

 a second deleting module 7225, configured to: if the index shard contains the directory metadata, determine the index shard as the second index shard and delete the index information corresponding to the directory metadata in the second index shard;

15 a third determining module 7226, configured to determine modified paths of the directory metadata and its subdirectory according to the final path and the initial path;

 a second creating module 7227, configured to create index information of a modified path in a corresponding second index shard according to the modified paths of the directory metadata and the subdirectory of the directory metadata.

20 **[0144]** The apparatus of this embodiment can perform the technical solutions of the above method embodiments, and the specific implementation processes and principles thereof are the same and will not be repeated here.

[0145] According to embodiments of the present disclosure, the present disclosure also provides an electronic device, a readable storage medium and a computer program product.

25 **[0146]** According to an embodiment of the present disclosure, the present disclosure also provides a computer program product. The computer program product includes a computer program, and the computer program is stored in a readable storage medium. At least one processor of an electronic device can read the computer program from the readable storage medium, and the at least one processor executes the computer program to cause the electronic

device to perform the solution provided in any of the foregoing embodiments.

[0147] FIG. 8 shows a schematic block diagram of an example electronic device 800 that may be used to implement embodiments of the present disclosure. The electronic device is intended to represent various forms of digital computers, such as a laptop computer, a desktop
5 computer, a workstation, a personal digital assistant, a server, a blade server, a mainframe computer, and other suitable computers. The electronic device may also represent various forms of mobile apparatuses, such as personal digital assistant, a cellular phone, a smartphone, a wearable device, and other similar computing apparatuses. Components, their connections and relationships, and their functions shown herein are merely examples, and are not intended to
10 limit the implementation of the present disclosure described and/or claimed herein.

[0148] As shown in FIG. 8, the device 800 includes a computing unit 801, which may perform various appropriate actions and processing according to a computer program stored in a read-only memory (ROM) 802 or a computer program loaded from a storage unit 808 to a random-access memory (RAM) 803. In the RAM 803, various programs and data required for
15 operations of the device 800 may also be stored. The computing unit 801, the ROM 802, and the RAM 803 are connected to each other by a bus 804. An input/output (I/O) interface 805 is also connected to the bus 804.

[0149] A number of components in the device 800 are connected to the I/O interface 805, including: an input unit 806, for example, a keyboard, a mouse, etc.; an output unit 807, for
20 example, various types of displays, speakers, etc.; a storage unit 808, for example, a disk, an optical disc, etc.; and a communication unit 809, for example, a network card, a modem, a wireless communication transceiver, etc. The communication unit 809 allows the device 800 to exchange information/data with other devices through a computer network such as the Internet and/or various telecommunication networks.

[0150] The computing unit 801 may be various types of general-purpose and/or special-purpose processing components with processing and computing capabilities. Some examples of the computing unit 501 include, but are not limited to, a central processing unit (CPU), a graphics processing unit (GPU), various dedicated artificial intelligence (AI) computing chips, various computing units for running machine learning model algorithms, a digital signal
25

processor (DSP), and any appropriate processor, controller, microcontroller, etc. The computing unit 801 executes the various methods and processing described above, for example, a method for processing directory metadata in a distributed file system. For example, in some embodiments, the method for processing directory metadata in a distributed file system may be implemented as a computer software program, which is tangibly contained in a machine-readable medium, for example the storage unit 808. In some embodiments, part or all of computer programs may be loaded into and/or installed on the device 800 via the ROM 802 and/or the communication unit 809. When the computer program is loaded into the RAM 803 and executed by the computing unit 801, one or more steps of the method for processing the directory metadata in the distributed file system described above may be performed. Alternatively, in other embodiments, the computing unit 801 may be configured to perform the method for processing the directory metadata in the distributed file system by means of any other appropriate means (for example, by means of firmware).

[0151] FIG. 9 is a schematic structural diagram of a distributed file system that can implement embodiments of the present disclosure. The distributed file system may include a plurality of device nodes. Each of the device nodes is set in an electronic device. As shown in FIG. 9, one device node of the distributed file system may include a metadata cell and an index cell. The metadata cell may include a plurality of metadata shards. Each metadata shard may correspond to a directory metadata. The index cell may include a plurality of index shards. The number of index shards may be set when the device node is created. Each index shard may store index information of a portion of directory metadata. In addition, the view of directory metadata and file metadata of the device node may be shown in the tree structure in FIG. 9. The box represents the directory metadata and the circle represents the file metadata, and the view may include a tree structure of all directory metadata and file metadata on the device node.

[0152] Various embodiments of the systems and the technologies described above herein may be implemented in a digital electronic circuit system, an integrated circuit system, a field programmable gate array (FPGA), an application-specific integrated circuit (ASIC), an application-specific standard product (ASSP), a system on chip (SOC), a complex programmable logic device (CPLD), computer hardware, firmware, software, and/or a

combination thereof. These various embodiments may include implementations in one or more computer programs, where the one or more computer programs may be executed and/or interpreted on a programmable system including at least one programmable processor, and the programmable processor may be a dedicated or general-purpose programmable processor that
5 may receive data and instructions from a storage system, at least one input apparatus and at least one output apparatus, and transmit data and instructions to the storage system, the at least one input apparatus and the at least one output apparatus.

[0153] Program codes used to implement the methods of the present disclosure may be written in any combination of one or more programming languages. These program codes may
10 be provided to a processor or a controller of a general-purpose computer, a special-purpose computer, or other programmable data processing devices, so that when the program codes are executed by the processor or the controller, functions/operations specified in flowcharts and/or block diagrams are implemented. The program codes may be entirely executed on a machine, partly executed on the machine, and as an independent software package, partly executed on
15 the machine and partly executed on a remote machine, or entirely executed on a remote machine or a server.

[0154] In the context of the present disclosure, a machine-readable medium may be a tangible medium, which may contain or store a program for use by an instruction execution system, apparatus, or device or for use in combination with the instruction execution system,
20 apparatus, or device. The machine-readable medium may be a machine-readable signal medium or a machine-readable storage medium. The machine-readable medium may include, but is not limited to, an electronic, magnetic, optical, electromagnetic, infrared or semiconductor system, apparatus or device, or any suitable combination of the foregoing. More specific examples of the machine-readable storage medium would include an electrical connection based on one or
25 more wires, a portable computer disk, a hard disk, a random-access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or flash memory), an optical fiber, a portable compact disc read-only memory (CD-ROM), an optical storage device, a magnetic storage device, or any suitable combination of the foregoing.

[0155] In order to provide an interaction with a user, the systems and the technologies

described herein may be implemented on a computer, the computer being equipped with: a display apparatus for displaying information to the user (for example, a CRT (cathode ray tube) or LCD (liquid crystal display) monitor); a keyboard and a pointing apparatus (for example, a mouse or a trackball) through which the user may provide input to the computer. Other types of apparatuses may also be used to provide an interaction with the user; for example, a feedback provided to the user may be any form of sensory feedback (for example, visual feedback, auditory feedback, or tactile feedback); and input from the user may be received in any form (including an acoustic input, a voice input, or a tactile input).

[0156] The systems and the technologies described herein may be implemented in a computing system (for example, as a data server) that includes a back-end component, or a computing system (for example, an application server) that includes a middleware component, or a computing system (for example, a user computer with a graphical user interface or a web browser through which the user may interact with the embodiments of the systems and the technologies described herein) that includes a front-end component, or a computing system that includes any combination of the back-end component, the intermediate component or the front-end component. Components of the system may be connected to each other through digital data communication (for example, a communication network) of any form or medium. Examples of the communication network include: a local area network (LAN), a wide area network (WAN), and the Internet.

[0157] A computer system may include a client and a server. The client and the server are generally far away from each other and usually interact through a communication network. A relationship between the client and the server is generated by computer programs that run on corresponding computers and have a client-server relationship with each other. The server may be a cloud server, also known as a cloud computing server or a cloud host, which is a host product in a cloud computing service system, to solve defects of high management difficulty and weak business scalability existing in services of a traditional physical host and a virtual private server (Virtual Private Server, or VPS for short) service. The server may also be a server of a distributed system, or a server combined with a blockchain.

[0158] It should be understood that steps may be reordered, added or deleted for various

forms of procedures shown above. For example, the steps recorded in the present disclosure may be performed in parallel, sequentially, or in a different order, as long as desired results of the technical solutions disclosed in the present disclosure can be realized, which is not limited herein.

- 5 **[0159]** The above specific embodiments do not constitute a limitation to the protection scope of the present disclosure. Those skilled in the art should understand that various modifications, combinations, sub-combinations and substitutions may be made according to design requirements and other factors. Any modification, equivalent replacement, improvement and so on made within the principle of the present disclosure shall be included in the protection
- 10 scope of the present disclosure.

CLAIMS

1. A method for processing directory metadata in a distributed file system, applied to a distributed file system, wherein a device node of the distributed file system is provided with a metadata cell containing a plurality of metadata shards and an index
5 cell containing a plurality of index shards, the method comprising:

in response to a processing request, determining (110) and loading a first index shard corresponding to directory metadata indicated by the processing request, and determining a metadata shard corresponding to the directory metadata indicated by the processing request according to the first index shard, wherein the processing request
10 represents an adjustment to directory metadata in a device node;

adjusting (120) information in the determined metadata shard, and adjusting index information corresponding to the directory metadata in a second index shard related to the directory metadata indicated by the processing request.

2. The method according to claim 1, wherein the processing request comprises
15 path information of the directory metadata;

wherein determining (110) and loading the first index shard corresponding to the directory metadata indicated by the processing request comprises:

calculating (210) an index code according to the path information of the directory metadata, wherein a value of the index code represents a code of the index shard;

20 loading (220) the first index shard corresponding to the directory metadata indicated by the processing request according to the index code.

3. The method according to claim 2, wherein when the processing request indicates deletion or creation of the directory metadata, the path information of the directory metadata comprises a target path; the target path indicates a path of the directory
25 metadata;

wherein calculating (210) the index code according to the path information of the directory metadata comprises:

calculating a first index code according to the target path;

determining a target parent path of a parent directory of the directory metadata in the target path according to the target path of the directory metadata;

calculating a second index code according to the target parent path, preferably,

calculating the first index code according to the target path comprises:

5 calculating an MD5 value of the target path;

determining the first index code according to the MD5 value and a preset number of the index shards in the index cell.

4. The method according to claim 2, wherein when the processing request indicates modification of a path of the directory metadata, the path information of the directory
10 metadata comprises an initial path and a final path; the initial path indicates a path of the directory metadata before the modification, and the final path indicates a path of the directory metadata after the modification;

wherein calculating (210) the index code according to the path information of the directory metadata comprises:

15 determining a first parent path of a parent directory of the directory metadata in the final path according to the final path of the directory metadata;

calculating a third index code according to the first parent path;

determining a second parent path of a parent directory of the directory metadata in the initial path according to the initial path of the directory metadata;

20 calculating a fourth index code according to the second parent path.

5. The method according to any one of claims 1-4, wherein the processing request comprises path information of the directory metadata;

25 wherein determining (110) the metadata shard corresponding to the directory metadata indicated by the processing request according to the first index shard comprises:

determining (310) a metadata code corresponding to the directory metadata in the first index shard according to the path information of the directory metadata indicated by the processing request, wherein the metadata code represents a code of the metadata shard;

determining (320) the metadata shard corresponding to the directory metadata indicated by the processing request according to the metadata code, preferably,

determining (310) the metadata code corresponding to the directory metadata in the first index shard according to the path information of the directory metadata indicated by the processing request comprises:

determining index information corresponding to the path information in the first index shard according to the path information of the directory metadata indicated by the processing request;

determining the metadata code corresponding to the index information in the first index shard.

6. The method according to any one of claims 1-5, wherein when the processing request indicates deletion of the directory metadata, the metadata shard comprises a first metadata shard corresponding to the directory metadata and a second metadata shard corresponding to a parent directory of the directory metadata;

wherein adjusting (120) the information in the determined metadata shard comprises:

deleting the first metadata shard;

subtracting a quantity of the directory metadata from information of the second metadata shard.

7. The method according to any one of claims 1-6, wherein when the processing request indicates creation of the directory metadata, the metadata shard comprises a first metadata shard corresponding to the directory metadata and a second metadata shard corresponding to a parent directory of the directory metadata;

wherein adjusting (120) the information in the determined metadata shard comprises:

creating the first metadata shard;

increasing a quantity of the directory metadata to information of the second metadata shard.

8. The method according to any one of claims 1-7, wherein when the processing

request indicates modification of a directory of the directory metadata, the metadata shard comprises a third metadata shard corresponding to a parent directory indicated by a final path of the directory metadata and a fourth metadata shard corresponding to a parent directory indicated by an initial path of the directory metadata;

5 wherein adjusting (120) the information in the determined metadata shard comprises:

 increasing a quantity of the directory metadata to information of the third metadata shard;

 subtracting a quantity of the directory metadata from information of the fourth
10 metadata shard.

9. The method according to any one of claims 1-8, wherein when the processing request indicates deletion of the directory metadata, path information of the directory metadata comprises a target path;

 wherein adjusting (120) the index information corresponding to the directory
15 metadata in the second index shard related to the directory metadata indicated by the processing request comprises:

 determining the second index shard according to the target path of the directory metadata;

 deleting the index information corresponding to the target path of the directory
20 metadata from the second index shard.

10. The method according to any one of claims 1-9, wherein when the processing request indicates creation of the directory metadata, path information of the directory metadata comprises a target path;

 wherein adjusting (120) the index information corresponding to the directory
25 metadata in the second index shard related to the directory metadata indicated by the processing request comprises:

 determining the second index shard according to the target path of the directory metadata;

 creating the index information corresponding to the target path of the directory

metadata in the second index shard.

11. The method according to any one of claims 1-10, wherein when the processing request indicates modification of a directory of the directory metadata, path information of the directory metadata comprises a final path and an initial path;

5 wherein adjusting (120) the index information corresponding to the directory metadata in the second index shard related to the directory metadata indicated by the processing request comprises:

10 if the index shard contains the directory metadata, determining the index shard as the second index shard and deleting the index information containing the directory metadata in the second index shard;

 determining modified paths of a subdirectory of the directory metadata and the directory metadata according to the final path and the initial path;

15 creating index information of a modified path in a corresponding second index shard according to the modified paths of the subdirectory of the directory metadata and the directory metadata.

12. An apparatus (600) for processing directory metadata in a distributed file system, applied to a distributed file system, wherein a device node of the distributed file system is provided with a metadata cell containing a plurality of metadata shards and an index cell containing a plurality of index shards, the apparatus comprising:

20 a determining unit (610), configured to, in response to a processing request, determine and load a first index shard corresponding to directory metadata indicated by the processing request, and determine a metadata shard corresponding to the directory metadata indicated by the processing request according to the first index shard, wherein the processing request represents an adjustment to directory metadata in a device node;

25 an adjusting unit (620), configured to adjust information in the determined metadata shard, and adjust index information corresponding to the directory metadata in a second index shard related to the directory metadata indicated by the processing request.

13. A distributed file system, comprising a plurality of device nodes, wherein the

device node is provided with a metadata cell and an index cell; the metadata cell comprises metadata shards corresponding to directory metadata in the device node, and the index cell comprises a plurality of index shards, each index shard is configured to store index information of a portion of the directory metadata;

5 wherein the device nodes are set in an electronic device configured to perform the method according to any one of claims 1-11.

14. A non-transitory computer-readable storage medium storing computer instructions, wherein the computer instructions are used to cause a computer to execute the method according to any one of claims 1-11.

10 15. A computer program product comprising a computer program, wherein the method according to any one of claims 1-11 is implemented when the computer program is executed by a processor.